UDC: 004.93 DOI: https://doi.org/10.15407/jai2022.01.288

HYBRID NEURAL NETWORKS FOR NOISE REDUCTIONS OF INTEGRATED NAVIGATION COMPLEXES

V. Sineglazov
V.M. Glushkov Institute of Cybernetics of the National Academy of Sciences of Ukraine
40 Academician Glushkov ave., Kyiv, Ukraine, 03187
svm@nau.edu.ua
https://orcid.org/0000-0002-3297-9060

Annotation. The necessity of integrated navigation complexes (INC) construction is substantiated. It is proposed to include in the complex the following inertial systems: inertial, satellite and visual. It helps to increase the accuracy of determining the coordinates of unmanned aerial vehicles. It is shown that in unfavorable cases, namely the suppression of external noise of the satellite navigation system, an increase in the errors of the inertial navigation system (INS), including through the use of accelerometers and gyroscopes manufactured using MEMS technology, the presence of bad weather conditions, which complicates the work of the visual navigation system. In order to ensure the operation of the navigation complex, it is necessary to ensure the suppression of interference (noise). To improve the accuracy of the INS, which is part of the INC, it is proposed to use the procedure for extracting noise from the raw signal of the INS, its prediction using neural networks and its suppression. To solve this problem, two approaches are proposed, the first of which is based on the use of a multi-row GMDH algorithm and single-layer networks with sigm_piecewise neurons, and the second is on the use of hybrid recurrent neural networks, when neural networks were used, which included long-term and short-term memory (LSTM) and Gated Recurrent Units (GRU). Various types of noise, that are inherent in video images in visual navigation systems are considered: Gaussian noise, salt and pepper noise, Poisson noise, fractional noise, blind noise. Particular attention is paid to blind noise. To improve the accuracy of the visual navigation system, it is proposed to use hybrid convolutional neural networks.

Keywords: integrated navigation complex, hybrid recurent neural network, hybrid convolutional neural network.

ГІБРИДНІ НЕЙРОННІ МЕРЕЖІ ДЛЯ ЗНИЖЕННЯ ШУМУ ІНТЕГРОВАНИХ НАВІГАЦІЙНИХ КОМПЛЕКСІВ

В. М. Синєглазов
Інститут кібернетики імені В. М. Глушкова Національної академії наук України
пр. Академіка Глушкова, 40, Київ, Україна, 03187
svm@nau.edu.ua
https://orcid.org/0000-0002-3297-9060

Анотація. Обґрунтовано необхідність побудови інтегрованих навігаційних комплексів (ІНК) у складі інерціальної, супутникової та візуальної навігаційних систем, що сприяє підвищенню точності визначення координат безпілотних літальних апаратів. Показано, що в несприятливих випадках, а саме придушення зовнішніми перешкодами супутникової системи навігації, наростання помилок інерціальної системи навігації (ІНС), в тому числі за рахунок застосування акселерометрів і гіроскопів, виконаних з використанням МЕМС технології, наявності поганих погодних умов, що ускладнюють роботу системи візуальної навігації. З метою забезпечення роботоспроможності навігаційного комплексу потрібно забезпечити придушення перешкод (шуму). Для підвищення точності ІНС, що входить до складу ІНК, пропонується використовувати процедуру виділення шуму з необробленого сигналу ІНС, його прогнозування за допомогою нейронних мереж та його придушення. Для вирішення цього завдання запропоновано два підхода, перший з яких основан на використанні багаторядного алгоритму МГУА та одношарових мереж з нейронами sigm_piecewise, а другий - на використанні гібридних рекурентних нейронних мереж, коли використовувались нейронні мережі,, до складу яких входять довготривала та короткочасна пам'ять та керований рекурентний блок (вентильний рекурентний вузол). Для підвищення точності системи візуальної навігації пропонується використовувати гібридні згорткові нейронні мережі. Розглянуто різні типи шумів,які притаманні відео зображенням в системах візуальної навігації:гаусів шум, шум солі та перцю, пуасонівський шум, дробовий шум, сліпий шум (blind noise). Особливу увагу приділено сліпим шумам (blind noise). Для підвищення точності системи візуальної навігації пропонується використовувати гібридні згорткові нейронні мережі.

Ключові слова: інтегрований навігаційний комплекс, гібридна рекурентна нейронна мережа, гібридна згорткова нейронна мережа.

Introduction

Requirements for the accuracy and reliability of navigation support are characterized by the following indicators: accuracy, readiness (the degree of probability of the complex operability), integrity (the degree of probability of detecting a failure within a given time) and reliability (the degree of probability of maintaining the required accuracy of the complex operation during the specified time intervals). In most cases, the reliability and quality of navigation support depends on the redundancy factor [1 – 3].

The general trend in the development of the market for navigation systems for mobile objects is such that, under the influence of increasingly stringent requirements, developers are moving towards deepening integration between inertial, satellite and other systems. At the same time, the ICAO Committee on Advanced Navigation Systems (FANS-Future Air Navigation System) recommends the use of onboard Satellite Navigation System (SNS) with a mandatory combination with an inertial navigation system (INS) as the central link of the navigation complex. This is due to the following reasons: INS provide complete information on navigation parameters of movement – angles of heading, pitch (trim), roll; acceleration, speed of movement and coordinates of the object's location. Moreover, they are completely autonomous, i.e. do not require any information from outside. Due to the ability to determine the angular position of an object with high accuracy in any range of angles and with a high frequency of information output, ANNs have no alternative to date. SNS – is an all-weather navigation system covering the earth, and its positioning accuracy does not diverge over time, provides precise navigation solutions at low cost, since a handheld chip receiver is cheap and sufficient for common applications.

However, in conditions of limited visibility of satellites, as well as in case of loss of information from a satellite navigation system, for example, due to difficult radio-technical conditions for receiving signals, there is not enough initial data to obtain an acceptable navigation solution. Therefore, in this case, there is a threat of loss of information support for the flight of the Unmanned Aerial Vehicles (UAV) and, as a consequence, of the loss of the UAV itself (the coarse Strapdown Inertial Navigation System (SINS) is not able to provide the UAV with flight and navigation information of the required accuracy, even at short intervals).

In this case, the integration system will turn into an autonomous inertial navigation system that provides navigation solutions. However, without outside help, the positioning errors of the inertial navigation system (INS) quickly diverge due to the noise contained in the initial data of the inertial measuring unit. In particular, the inertial measuring unit of the micromechanics system allows more complex errors due to the manufacturing technology.

Thus, each of the above systems separately cannot provide a high-quality solution to the navigation problem. Therefore, it is advisable to build an integrated navigation complex (INC), which includes navigation systems operating according to various physical principles, which will make it possible to increase the reliability of navigation support for UAV flight, both in basic and in autonomous modes of operation. Increasing the reliability (credibility) of navigation support can be achieved at the expense of the maximum, including the alternative. using the capabilities of the existing information support, as well as through new approaches to the problems of integrating flight and navigation information [4]. The visual navigation system (VNS), which consists of a built-in microcomputer and a camera, is free from the above disadvantages, which is advisable to include in the INC. Image processing is realized by an intelligence system based on convolution neural networks.

There are many ways to combine information from the many navigation systems that make up the INC. Designing an integration architecture is a trade-off between

increased accuracy and robustness in obtaining a navigation "solution" that minimizes complexity and optimizes the efficiency of navigation information processing. This should also be taken into account for the navigation systems that make up the INC.

But in unfavorable cases, namely suppression by external interference of the satellite navigation system, an increase in errors of the inertial navigation system, including due to the use of accelerometers and gyroscopes made using MEMS technology, the presence of bad weather conditions that impede the operation of the VNS, with in order to ensure the operation of the navigation complex, it is required to provide interference suppression.

To improve the accuracy of INS, which is part of the INC, it is proposed to use the procedure of noise separation from INS raw signal, its predicting with help of neuron networks and its suppression. The procedure of noise separation from INS raw signal is considered in [5]. The gyroscope data was processed as a time series, and the experiments used a real data set from the inertial measuring unit of the micromechanical system.

Problem Statement of nonstationary processes predicting

For a class of nonlinear nonstationary processes:

$$\{X(k,\varepsilon)\} \in \Re^N,$$

where $E[X(k)] \neq \text{const}, \quad \text{var}[X(k)] \neq \text{const},$ build predictive models (neural networks)

$$x(k) = f[\theta, z(k), \varepsilon(k)],$$

where θ is the vector of model parameters; $k = 0, 1, 2,$ discrete time; $z(k)$ is the independent explanatory variables; $\varepsilon(k)$ is the random disturbing process with arbitrary distribution.

Based on the model (in the form of a neural network) to create the prediction features

$$\hat{x}(k+s) = E_k[x(k+s)|x(k), x(k-1),..., x(0), \hat{\varepsilon}(k),..., \hat{\varepsilon}(0)],$$

where $x(k+s)$ is the a function that allows you to calculate future values of a master (dependent) variable based on known historical data $\{x(k), x(k-1), ..., x(0)\}$ and estimates of random process values $\{\hat{\varepsilon}(k), \hat{\varepsilon}(k-1), ..., \hat{\varepsilon}(0)\}$; s is the number of forecasting steps.

Analyze the quality of forecast estimates by many of the following statistical criteria: MSE error and average absolute percentage error (MAPE):

$$\text{MAPE} = \frac{1}{s}\sum_{i=1}^{s}\frac{\left|x_f(k+i) - \hat{x}(k+i)\right|}{\left|x_f(k+i)\right|}100\%,,$$

where $x_f(k+i)$ is the actual values of the main variable pertaining to the test sample; $\hat{x}(k+i)$ are estimates.

Predicting time series using neural networks

After analyzing the various methods currently available for constructing a predictive model, the most flexible methods can be clearly called the Artificial Neural Networks (ANN) and the Group method of data handling (GMDH). In this work it is proposed to use two approaches. The first is based on combination of ANN and GMDN, where the ANN is used as a reference function of the GMDH. Due to this, this algorithm combines the advantages of both ANN and GMDH: a gradual increase in the complexity of the GMDH model and the opportunity to learn GMDH. The second approach is based on hybrid recurrent neural network (RNN) which is a combination of Long Short Term Memory (LSTM) and the Gated Recurrent Unit (GRU) [6].

The use of the GMDH multilayer algorithm and single-layer networks with sigm_piecewise neurons

To solve all these problems, it is proposed to use the approach of the GMDH multilevel algorithm [7], which will allow:

– much faster than under full brute force use, to find in a certain sense the optimal network structure, with given restrictions on the number of network parameters;

– automatically filter out "uninformative" variables.

The algorithm for automatic synthesis of optimal network structure with sigm_piecewise neurons based on the GMDH multilevel algorithm consists of the following steps:

1) Input sample $< X, y : X \to R >$, where the set x is a finite subset of space $R^n, n > 2$ is in some way divided into two samples: training R^n, $n > 2$ and validational – $< X^{(V)}, y^{(V)} : X^{(V)} \to R >, X^{(V)} \subset X$, $X^{(V)} \cap X^{(T)} = \varnothing, X^{(T)} \cup X^{(V)} = X$. Usually, about 70% of the patterns are selected in the training sample, and the simplest variant, which works well in practice, is random selection.

2) Independently one from one C_n^k neurons of the sigm_piecewise type are learing, where each neuron has inputs – $x_i, x_i;\ i, j = 1,...,n, i < j$ – that is, all possible pairs of inputs are sorted out, and a separate neuron is trained for each pair. For learning it is used training sample, the criterion according to which the parameters of each neuron are tuned is the following:

$$E_{i,j}(\vec{w}_+, \vec{w}_-, \vec{h}) = \sum_{\vec{x} \in X^{(T)}} \left(y^{(T)}(\vec{x}) - \text{sigm\_piecewise}([x_i, x_j]^T; (\vec{w}_+, \vec{w}_-, \vec{h}) \right)^2.$$

3) The value of the "external" criterion is calculated for each neuron. The most common criterion is the MSE model on the validation sample:

$$C_{i,j}(\vec{w}_+, \vec{w}_-, \vec{h}) = \sum_{\vec{x} \in X^{(V)}} \left(y^{(V)}(\vec{x}) - \text{sigm\_piecewise}([x_i, x_j]^T; \vec{w}_+, \vec{w}_-, \vec{h}) \right)^2.$$

4) $\lfloor \alpha \cdot C_k^n \rfloor, \alpha \in (0,1)$ the neurons with the worst external criterion value are discarded. Parameter value α is selected depending on the limit on the number of network parameters, or according to some heuristic, if no such restriction. For example, if there are variables in the input variables that are completely unrelated to the predicted variable, then the external criterion value for the neurons that used these variables as inputs is usually much worse than the value for the neurons that used the "informative" variables (Fig. 1).

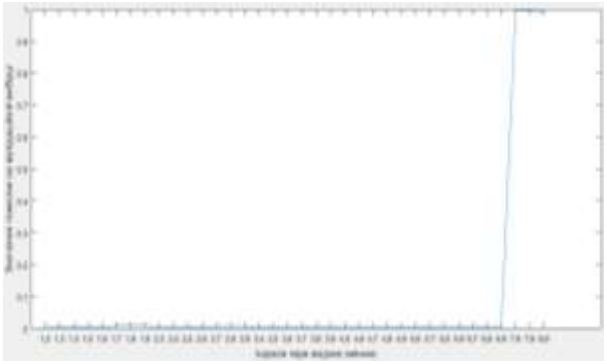

Fig. 1. An example of the values of external criterion "model error on validation sample" of different neurons with the presence of "uninformative" variables

Another approach of MEMS sensors noise prediction is the use of recurrent neural networks, in particular long short-term memory (LSTM), gated recurrent units (GRU) and a hybrid neural network consisting of LSTM and GRU [8].

Let's first consider each of the LSTM and GRU networks separately.

Description of LSTM

Thus, the LSTM was proposed to solve the long-term dependence problem of the existing RNN. It is a modified architecture of the RNN and is characterized by the existence of a cell state (Fig. 2). The LSTM is composed of three gates, namely the input, forget, and output gates. Each gate controls its value through the sigmoid layer and pointwise multiplication [8].

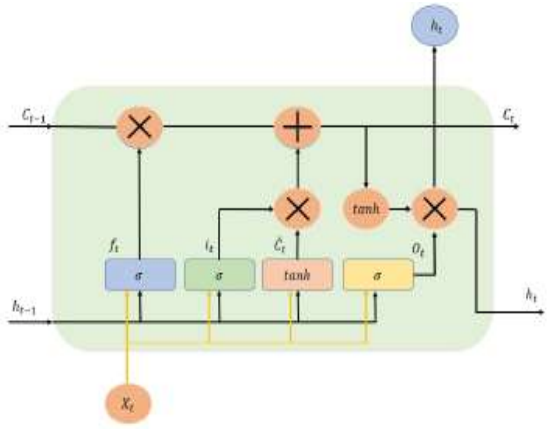

Fig. 2. Long short-term neural network architecture

The input gate (it) utilizes a sigmoid function (σ). The sigmoid function determines

which of the new information is to be stored in the cell input state. Next, the tanh function creates new data candidate values as the cell candidate state, $\tilde{C}_t$.

The forget gate determines what is forgotten using the weights of how much to forget and updates the state by multiplying the previous state, C_{t-1}, by the output value of the forget gate. In addition, the output values of the input gate are multiplied and added to the cell state C_t.

The output gate (O_t) determines the output value and uses a sigmoid function. The final output value is calculated by multiplying the value obtained by applying tanh to the cell state (C_t) and the sigmoid output value (O_t).

The LSTM unit is the most basic component of an LSTM architecture; it's a series of gates and cells that work together to produce a final result. The forward pass of an LSTM unit is modeled by follows equatios [9].

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right), \quad (1)$$

$$i_t = \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right), \quad (2)$$

$$\tilde{c}_t = \tanh\left(W_c x_t + U_c h_{t-1} + b_c\right), \quad (3)$$

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t, \quad (4)$$

$$o_t = \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right), \quad (5)$$

$$h_t = o_t * \tanh\left(c_t\right), \quad (6)$$

where σ is sigmoid function, f_t is forget gate activation vector, i_t input gate activation vector, o_t output gate activation vector, $\tilde{c}_t$ cell input activation vector, c_t cell state, h_t output vector of LSTM unit, all W and U are weights, b is biases vector and symbol * for Hadamard product (element wise product). Weights W, U and biases b are to be learned during the training process. In order to better decipher the equations cited above, let's start with cell state c_t, it contains two kinds of information: old information to keep from past state c_{t-1} specified using forget gate, it decides the percentage of information to keep by computing a value between 0 (throw completely) and 1 (keep completely), and new information to include in the cell state calculated using input gate i_t and cell activation $\tilde{c}_t$ which are computed using (2) and (3), respectively. Finally the calculation of the fnal output value is performed in two steps. A potential value is calculated using (5) for the first time. This value will be regulated using the information present in cell state as indicated in (6). Te use of the cell state in the final calculation makes the LSTM powerful in tasks where information must be stored and used later (long term).

Description of GRU

Although the LSTM has an excellent ability to memorize long-term dependencies, it takes a long time to train the model owing to its complex structure. Thus, the GRU was proposed to accelerate the training process [10].

The GRU is a type of RNN framework with a gate mechanism applied, inspired by LSTM, and with a simpler structure [11]. The GRU architecture is described in Fig. 3.

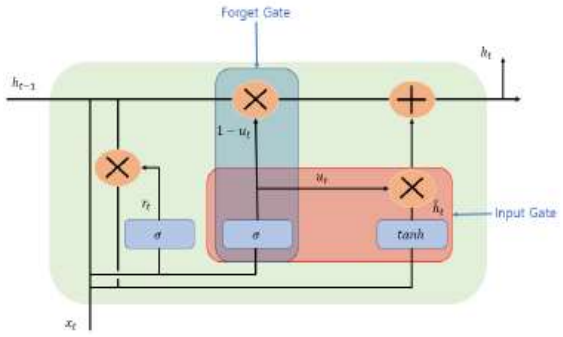

Fig. 3. Gated recurrent unit architecture

The reset gate (r_t) uses the sigmoid function as an output to properly reset the past information and multiplies the (0.1) value by the previously hidden layer. The output value can be obtained by multiplying the hidden layer's value at the previous point and the information at the present point by weight.

The update gate is a combination of the forget and input gates of the LSTM model. The update gate determines the rate of update of past and present information. In the update gate, the result u_t output as sigmoid determines the amount of information at present, and the value subtracted from $1(1-u_t)$ is multiplied by the information of the hidden layer at the most recent time. Each update gate is similar to the input and forget gates of the LSTM.

The forward pass of a GRU unit is modeled by follows equatios [9]:

$$z_t = \sigma\left(W_z x_t + U_z h_{t-1} + b_z\right), \quad (7)$$

$$r_t = \sigma\left(W_r x_t + U_r h_{t-1} + b_r\right), \quad (8)$$

$$\tilde{h}_t = \tanh\left(W_h x_t + U_h\left(r_t * h_{t-1}\right) + b_h\right), \quad (9)$$

$$h_t = \left(1 - z_t\right) * h_{t-1} + z_t * \tilde{h}_t, \quad (10)$$

where σ is sigmoid function, z_t is update gate activation vector, r_t reset gate activation vector, $\tilde{h}_t$ candidate vector and h_t is the output vector of GRU unit. *W* and *U* are weights, *b* is bias vector and symbol * for Hadamard product. Same as for LSTM, weights and biases are to be learned during the training process. Let us go deep through the equations cited above to better understand how a GRU unit works. First update gate is computed using input vector x_t, output of previous unit h_t −1 and sigmoid activation function. Reset gate is calculated in the same way as update gate using its own weights and bias. The reset gate is then involved in the candidate value calculation. It determines how much information from the previous state should be preserved. Indeed, from equation (9) we can notice that with an r_t value very close or equal to zero, only the input value will be considered in candidate value calculation. Finally, the output value is calculated by calibrating the previous output and the new candidate output. Te calibration is carried out by the updating gate z_t (z_t = 0 copy the previous output, z_t = 1 generate a new output regardless of the old output). We can observe a kind of similarity between LSTM and the GRU, indeed both implement an intermediate gating mechanism used later in the output value calculation.

It maybe noticed that the GRU optimization is faster. In general, both LSTM and GRU are powerful and well-suited to sequential data. The GRU has the beneft of faster optimization compared with an LSTM because it has less parameters.

Hybrid Neural Structure of LSTM and GRU

In practical applications, instead of using single-layer LSTM or GRU, multilayer LSTM or GRU is generally used however with no more than three layers [6]. The two-layer hybrid structure is shown in Fig. 4. The input layer was LSTM while the output layer was GRU and thus, a heterogeneous network abbreviated as LSTM-GRU was developed, while in Fig. 5, the input layer was GRU and the output layer was LSTM to produce the other heterogeneous network referred to as GRU-LSTM. In paper [6] the four hybrid models of deep RNN designed above all composed of four parts is presented. The first part was an initial state where in random initialization were made, while the second part was used for feeding the sequence data as networks' inputs. The third stage includes a collection of hidden states in each of the LSTM or GRU layer, while the last part is where the network prediction output is done. The latter can take the state output of the last step by weighing the state from all previous steps or directly averaging them to produce the output.

As mentioned earlier, the third component of the INC is the visual navigation system, which uses the processing of images obtained from a video camera to determine the current coordinates of the UAV. In case of bad weather conditions, the images obtained from the video camera will be of bad quality, which ultimately will lead to significant errors in determining the coordinates of the UAV. To overcome these difficulties, it is necessary to perform image noise reduction.

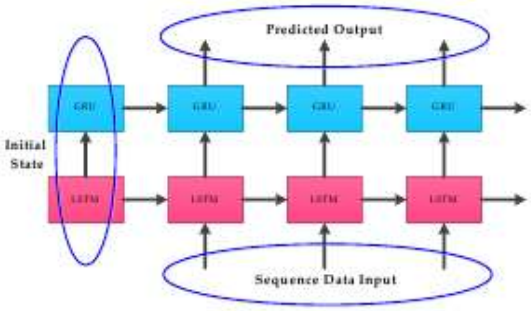

Fig. 4. The structure of LSTM-GRU

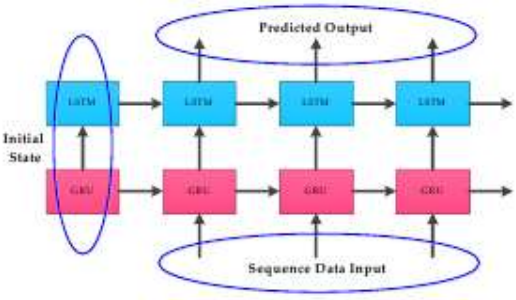

Fig. 5. The structure of GRU-LSTM

Deep Learning on Image Denoising

For today, the most promising approach to eliminating for image denoisimg is deep learning neural networks.

It is necessary to consider: deep convolutional neural networks (CNNs) for additive white noisy images, deep CNNs for real noisy images, deep CNNs for blind denoising and deep CNNs for hybrid noisy images, which is the combination of noisy, blurred and low-resolution images. Among deep learning networks for solving image noise reduction problems, convolutional neural networks are the most promising [12, 13]

However, due to the following drawbacks, they were not widely applied into computer systems [15]. Firstly, deep CNNs can generate vanishing gradients. Secondly, activation functions (i.e. sigmoid [16] and tanh [17]) resulted in high computational cost. Thirdly, the hardware platform did not support the complex network. That was broken by AlexNet in 2012 ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) [15]. After that, deep network architectures (e.g. VGG [18] and GoogleNet [19]) were widely applied in fields of image [20, 21] video [22, 23], nature language processing [24] and speech processing [25], especially low-level computer vision [26, 27].

Deep network was first applied in image denoising in 2015 [28, 29]. The proposed network need not manually set parameters for removing the noise. After then, deep network were widely applied in speech [30] and image restoration [31]. Mao et al. [31] used multiple convolutions and deconvolutions to suppress the noise and recover the high-resolution image. For addressing multiple low-level tasks via a model, a denoising CNN (DnCNN) [32] consisting of convolutions, batch normalization (BN) [33], rectified linear unit (ReLU) [34] and residual learning (RL) [35] was proposed to deal with image denoising, super-resolution, and JPEG image deblocking. Taking into account between denoising performance and speed, a color non-local network (CNL-Net) [36] combined non-local self-similarity (NLSS) and CNN to efficiently remove color-image noise.

In terms of blind denoising, a fast and flexible denoising CNN (FFDNet) [37] presented different noise levels and the noisy image patch as input of denoising network to improve denosing speed and process blind denoising.

Problem Statement of image denoising

The given image determined as

$$y = x + \mu,$$

where x, y and μ represent the given clean image, noisy image and additive Gaussian noise (AWGN) of standard deviation σ, respectively. The purpose of image noise reduction is to obtain a clear image x from a damaged version y.

Consider the following as additive image noises: Gaussian noise, salt and pepper noise, Poisson noise, shot noise. Consider each of the above noises separately.

Gaussian noise

The main sources of Gaussian noise in digital images occur during acquisition, such as sensor noise caused by poor lighting and/or high temperature, and/or transmission, such as electronic circuit noise. The standard model of this noise is additive, independent at each pixel, and independent of signal intensity, primarily caused by Johnson-Nyquist noise (thermal noise), including capacitor reset noise. Amplifier noise is the main part of the image sensor's read noise, i.e. constant noise level in dark areas of the image. In color cameras, where more gain is used in the blue channel than in the green or red channel, there may be more noise in the blue channel.

The noise of salt and pepper

The common "fat tail" or "impulsive" noise is sometimes referred to as salt and pepper noise or spike noise. An image containing salt and pepper noise will have dark pixels in the light areas and bright pixels in the dark areas. This type of noise can be caused by A/D converter errors, bit transmission errors, etc. This can be mostly eliminated using dark frame subtraction and interpolation around dark/bright pixels.

Poisson noise

Poisson noise or fractional noise is a

type of electronic noise that occurs when a finite number of energy-carrying particles, such as electrons in an electronic circuit or photons in an optical device, are small enough to cause statistical fluctuations in a measurement.

Shot noise

The predominant noise in the lighter parts of the image from the image sensor is usually the cause of statistical quantum fluctuations, i.e. by changing the number of photons that are perceived at a given level of exposure. This noise is known as photon shot noise. Shot noise has an RMS value proportional to the square root of the intensity of the image, and the noise in different pixels is independent of each other. Shot noise follows a Poisson distribution and is usually not very different from a Gaussian.

Statement of the problem in the presence of blind noise

The image is often distorted due to the imperfection of optical devices, defocusing of images, due to the influence of the environment between the object and the device for a number of other reasons. The problem of image restoration is, as a rule, incorrect (highly unstable) and it is necessary to use modern, stable methods to solve it. The solution of this problem requires the use of quite diverse and complex tools, including mathematical modeling, statistical estimation theory, Fourier transform, optimization methods, numerous linear algebra methods, etc. We will consider only halftone black and white images under the assumption that for processing a full-color image it is enough to repeat all the necessary steps for each each of the RGB color channels. Let us introduce the following notation: $f(x, y)$ – original undistorted image, $h(x, y)$ – distortion function, $n(x, y)$ – additive noise, $g(x, y)$ – result of distortion, i.e. what we see as a result (a blurry or out-of-focus image). We formulate the model of the distortion process as follows

$$g(x, y) = h(x, y) * f(x, y) + n(x, y).$$

The problem of reconstructing a distorted image is to find the best approximation $f'(x, y)$ of the original image. The function $h(x, y)$ determines the process of distortion (transformation) of each pixel of the original image into a spot for the case of defocusing and into a segment for simple blurring. Or you can say the opposite, each pixel of the distorted image is collected from the pixels of some neighborhood of the original image. All this is superimposed on each other and as a result we get a disfigured image.

Deep learning techniques for additive white noisy-image denoising

Due to insufficiency of real noisy images, additive white noisy images (AWNI) have been widely used to train the denoising model [38]. The AWNI includes Gaussian, Poisson, Salt, Pepper and multiplicative noisy images [39]. Moreover, there are a lot of deep learning techniques (i.e. CNN/NN, the combination of CNN/NN and common feature extraction methods and the combination of optimization method and CNN/NN) for AWNI denoising [3, 40 – 51].

The combination of optimization method and CNN/NN for AWNI denoising

It is known that machine learning uses optimization techniques [52, 53] and discriminative learning methods [54, 55] to deal with image applications in general. Although optimization methods have good performance on different low-level vision tasks, these methods need manual setting parameters, which were time-consuming. The discriminative learning methods have fast speed in image restoration. However, they are not flexible for various low-level vision tasks.

To make a tradeoff between efficiency and flexibility, discriminative learning optimization-based method [56] was presented for image applications, such as image denoising. The CNN with prior knowledge via regular term of loss function is common method in image denosing [57], which can mainly divide two categories to filter the noise:

1) improvement of denoising speed;
2) improvement of denoising performance.

For improving denoising speed, optimization method cooperated CNN was a good tool to rapidly find optimal solution in image denoising [58]. For example, a GAN

with maximum a posteriori (MAP) was used to estimate the noise and deal with other tasks, such as image inpainting and super-resolution [59]. An experience-based greedy and transfer learning strategies with CNN can accelerate genetic algorithm to obtain a clean image [60]. Noisy image and noise level mapping were as inputs of CNN, which had faster execution in predicting the noise [61].

For improving the denoising performance, CNN combined optimization methods to make noisy image smooth [62].

The main focus of real applications on deep learning for image denoising has two kinds: single end-to-end CNN and the combination of prior knowledge and CNN.

For the first method, changing the network architecture is popular to remove the noise from the given real corrupted image. Multiscale is very effective for image denoising. For example, a CNN comprising of convolution, ReLU and RL employed different phase features to enhance the expressive ability of the low-light image denoising model [63]. To overcome the blurry and false image artifacts, a dual U-Net with skip connection was proposed for CT image reconstruction [64]. To address resource-constraint problem, Tian et al. [65] used a dual CNN with batch renormalization [66], RL and dilated convolutions to deal with real noisy image. According to the nature of light image, two CNNs utilized anisotropic parallax analysis to generate structural parallax information for real noisy images [67]. Additionally, using CNN to resolve remote sense [68] and medical images under low-light condition is very effective. To extract more detailed information, recurrent connections were used to enhance the representative ability to deal with corrupted image in the real world [69]. To deal with unknown real noisy images, a residual structure was utilized to facilitate low-frequency features, then, an attention mechanism can be applied to extract more potential features from channels. From the point of view of producing the noisy image, imitating cameral pipelines to construct the degradation model was very effective to filter the real noisy [71].

Deep learning techniques for blind denoising

In the real world, the image is easily corrupted and noise is complex. Thus, blind denoising technique is important [72]. At first, FFDNet [37] used noise level and noise as the input of CNN to train a denoiser for unknown noisy image. Then, scholars proposed a lot of methods to solve blind denoisng problem. According to the mechanism of image device, Kenzo et al. [70] utilized soft shrinkage to adjust the noise level for blind denoising. For unpaired noisy image, using CNNs to estimate noise became a good tool [73]. Yang et al. [74] used known noise level to train a denoiser, then, they utilized this denoiser to estimate the level of noise. For random noise attenuation problem, CNN with RL was used to filter complex noise [75, 76]. Additionally, changing network architecture can promote the denoising performance for blind denoising.

In this work, to solve the problem of noise reduction of images obtained from a video camera of a visual navigation system, the following criterion was used

$$L_{\phi_i}\left(\hat{I}, I_{gt}\right) = \frac{1}{hwd}\left\|\phi_i(\hat{I}) - \phi_i(I_{gt})\right\|^2, \quad (11)$$

where $\hat{I}$ and I_{gt} – noise-free image and corresponding noise-free image $\left\{\phi_i(I)\middle| i = 1,...,N\right\}$ denote N different maps of image features I.

In this paper, we tackle this issue by developing a hybrid convolutional neural network. It is a blind denoising network (CBDNet) for real-world video. Ther CBDNet achieves very pleasing denoising results by retaining most structure and details while removing the sophisticated real-world noise. To sum up, the contribution of this work is four-fold:

– synthetic noisy images and real noisy video are incorporated for better characterizing real-world image noise and improving denoising performance;

– benefited from the introduction of noise estimation subnetwork, asymmetric loss is suggested to improve the generalization ability to real noise, and interactive denoising is allowed by adjusting the noise level map.

Blind denoising of real noisy images generally is more challenging and can involve two stages, i.e., noise estimation and non-blind denoising. In most blind denoising methods, noise estimation is closely coupled with non-blind denoising. CBDNet [77] consists of a noise estimation subnetwork CNN_E and a non-blind denoising subnetwork CNN_D. We let the output of CNN_E be the noise level map due to that it is of the same size with the input y and can be estimated with a fully convolutional network. Then, CNN_D takes both y and $\hat{\sigma}(\mathbf{y})$ as input to obtain the final denoising result $\hat{\mathbf{x}} = F_D(\mathbf{y}, \hat{\sigma}(\mathbf{y}); \mathbf{W}_D)$, where $\mathbf{W}_D$ denotes the network parameters of CNN_D [77].

Results

The use of the hybrid convolutional neural network CBDNet was tested when processing video images of cars in bad weather conditions (fog, light rain) on a sample of 100 images. The result of image processing is shown in Fig. 6.

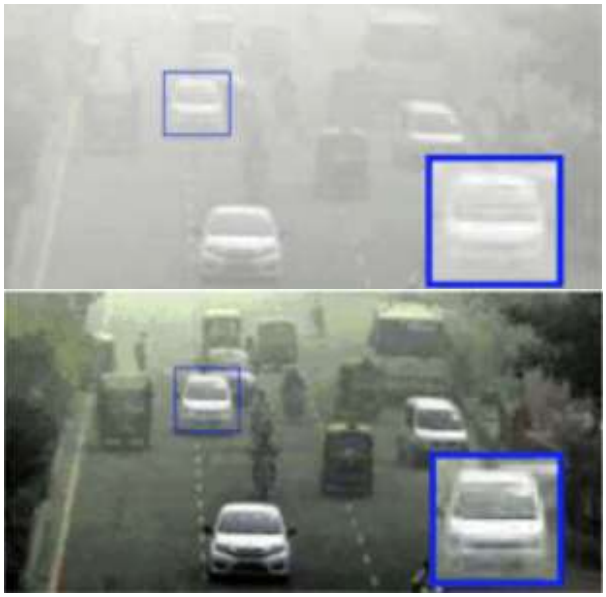

Fig. 6. The result of cars video image processing in bad weather conditions

Conclusions

The paper proposes approaches to improve the accuracy of integrated navigation systems by means of noise reduction methods use for inertial navigation and visual navigation systems. The problem is solved by using hybrid recurrent neural networks and hybrid convolutional neural networks.

References

1. Askerov Sh. Y. (2013) «Yntehryrovannыi navyhatsyonnыi kompleks bespylotnoho letatelnoho aparata,» Avtomatyzovane proektuvannia pilotazhno-navihatsiinykh kompleksiv bezpilotnykh litalnykh aparativ, Materialy KhI Mizhnar. nauk.-tekhn. konf. «Avia–2013» (21–23 travnia) – K.: NAU. – T. 4 – S. 21.5–21.9.
2. Askerov Sh. I. (2012) «Kompleksne obroblennia danykh u navihatsiinykh systemakh bespilotnykh litalnykh aparativ,» Elektronika ta systemy upravlinnia, № 1(31) – K.: NAU, S. 72–75.
3. Sineglazov, V. M., Askerov, Sh. I., Aksani, A. R. (2013) “Landmarks navigation system for unmanned aerial vehicles,” Electronics and Control Systems, no. 4(38). Kyiv: NAU, pp. 108–113. https://doi.org/10.18372/1990-5548.38.7294
4. Synehlazov V. M., Fyliashkyn N. K., Askerov Sh. Y., «Pryntsypы kompleksyrovanyia dannыkh v navyhatsyonnыkh systemakh BPLA», (2011) Materialy nauk.-tekhn. konf. «Aktualni problemy rozvytku bezpilotnykh litalnykh aparativ» (17–18 zhovtnia) – K.: NAU. – S. 105–109.
5. Filyashkin M. K. “Automation of technology of air-to-air refueling regional aircraft,” Electronics and Control Systems, N 2(48) – Kyiv: NAU, 2016. – pp. 87–91. https://doi.org/10.18372/1990-5548.48.11214.
6. Shipeng Han, Zhen Meng, Xingcheng Zhang and Yuepeng Yan, (2021) “Hybrid Deep Recurrent Neural Networks for Noise Reduction of MEMS-IMU with Static and Dynamic Conditions Micromachines,” 12, 214, r. 390. https://doi.org/10.3390/mi12020214 https://www.mdpi.com/journal/micromachines.
7. Michael Z. Zgurovsky, Viktor M. Sineglazov, Olena I. Chumachenko, Artificial Intelligence Systems Based on Hybrid Neural Networks, Springer, 2020, https://link.springer.com/book/10.1007/978-3-030-48453-8. Customer can order it via https://www.springer.com/gp/book/9783030484521
8. Jeong, M.-H., Lee, T.-Y., Jeon, S.-B., Youm, M. (2021) “Highway Speed Prediction Using Gated Recurrent Unit Neural Networks,” Appl. Sci., 11, 3059. https://doi.org/10.3390/app11073059.
9. Sehovac, Ljubisa, Nesen, Cornelius, and Grolinger, Katarina, (2019) "Forecasting Building Energy Consumption with Deep Learning: A Sequence to Sequence Approach" Electrical and Computer Engineering Publications. 166. https://ir.lib.uwo.ca/electricalpub/166.
10. Fu, R., Zhang, Z., and Li, L. (2016) “Using LSTM and GRU neural network methods for traffic flow prediction,” In Proceedings of the IEEE 31st Youth Academic Annual Conference of Chinese Association of Automation, Wuhan, China, 11–13 November 2016, pp. 324–328.
11. Cho, K., Van Merrilnboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and

Bengio, Y. (2014) Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv 2014, arXiv:1406.1078.
12. Lo, S.-C., Lou, S.-L., Lin, J.-S., Freedman, M. T., Chien, M. V., Mun, S. K., (1995) "Artificial convolution neural network techniques and applications for lung nodule detection," IEEE Transactions on Medical Imaging 14 (4), 711–718.
13. Li, J., Fang, F., Mei, K., Zhang, G., (2018) "Multi-scale residual network for image super-resolution," In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 517–532.
14. Jarrett, K., Kavukcuoglu, K., Ranzato, M., LeCun, Y., (2009) "What is the best multi-stage architecture for object recognition?" In: 2009 IEEE 12th international conference on computer vision. IEEE, pp. 2146–2153.
15. Krizhevsky, A., Sutskever, I., Hinton, G. E., (2012) "Imagenet classification with deep convolutional neural networks," In: Advances in neural information processing systems, pp. 1097–1105.
16. Marreiros, A. C., Daunizeau, J., Kiebel, S. J., Friston, K. J., (2008) "Population dynamics: variance and the sigmoid activation function," Neuroimage 42 (1), 147–157.
17. Simonyan, K., Zisserman, A., (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
18. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., (2015) "Going deeper with convolutions," In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–9.
19. Wang, H., Wang, Q., Gao, M., Li, P., Zuo, W., (2018) "Multi-scale location-aware kernel representation for object detection," In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1248–1257.
20. Wu, S., Xu, Y., (2019) "Dsn: A new deformable subnetwork for object detection," IEEE Transactions on Circuits and Systems for Video Technology.
21. Liu, Q., Lu, X., He, Z., Zhang, C., Chen, W.-S., (2017) "Deep convolutional neural networks for thermal infrared object tracking," Knowledge-Based Systems 134, 189–198.
22. Li, X., Liu, Q., Fan, N., He, Z., Wang, H., (2019) "Hierarchical spatial-aware siamese network for thermal infrared object tracking," Knowledge-Based Systems, 166, 71–81.
23. Duan, C., Cui, L., Chen, X., Wei, F., Zhu, C., Zhao, T., (2018) "Attention-fused deep matching network for natural language inference," In: IJCAI. pp. 4033–4040.
24. Zhang, Z., Geiger, J., Pohjalainen, J., Mousa, A. E.-D., Jin, W., Schuller, B., (2018) "Deep learning for environmentally robust speech recognition: An overview of recent developments," ACM Transactions on Intelligent Systems and Technology (TIST), 9 (5), 49.
25. Peng, Y., Zhang, L., Liu, S., Wu, X., Zhang, Y., Wang, X., (2019) "Dilated residual networks with symmetric skip connection for image denoising," Neurocomputing 345, 67–76.
26. Tian, C., Xu, Y., Fei, L., Wang, J., Wen, J., Luo, N., (2019) "Enhanced cnn for image denoising," CAAI Transactions on Intelligence Technology, 4 (1), 17–23.
27. Tian, C., Xu, Y., Fei, L., Wang, J., Wen, J., Luo, N., (2019) "Enhanced cnn for image denoising," CAAI Transactions on Intelligence Technology, 4 (1), 17–23.
28. Liang, J., Liu, R., (2015) "Stacked denoising autoencoder and dropout together to prevent overfitting in deep neural network," In: 2015 8th International Congress on Image and Signal Processing (CISP). IEEE, pp. 697–701.
29. Xu, Q., Zhang, C., Zhang, L., (2015) "Denoising convolutional neural network," In: 2015 IEEE International Conference on Information and Automation. IEEE, pp. 1184–1187.
30. Zhang, Z., Wang, L., Kai, A., Yamada, T., Li, W., Iwahashi, M., (2015) "Deep neural network-based bottleneck feature and denoising autoencoder-based dereverberation for distant-talking speaker identification," EURASIP Journal on Audio, Speech, and Music Processing 2015 (1), 12.
31. Mao, X., Shen, C., Yang, Y.-B., (2016) "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," In: Advances in neural information processing systems. pp. 2802–2810.
32. Zhang, K., Zuo, W., Chen, Y., Meng, D., Zhang, L., (2017) "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," IEEE Transactions on Image Processing, 26 (7), 3142–3155.
33. Ioffe, S., Szegedy, C., (2015) "Batch normalization: Accelerating deep network training by reducing internal covariate shift," arXiv preprint arXiv:1502.03167.
34. Nair, V., Hinton, G. E., 2010. Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th international conference on machine learning (ICML-10). pp. 807–814.
35. He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778.
36. Lefkimmiatis, S., (2017) "Non-local color image denoising with convolutional neural networks," In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3587–3596.
37. Zhang, K., Zuo, W., Zhang, L., (2018) "Ffdnet: Toward a fast and flexible solution for cnn-based image denoising," IEEE Transactions on Image Processing 27 (9), 4608–4622.
38. Jin, K. H., McCann, M. T., Froustey, E., Unser, M., (2017) "Deep convolutional neural network for inverse problems in imaging," IEEE Transactions on Image Processing 26 (9), 4509–4522.

39. Farooque, M. A., Rohankar, J. S., (2013) "Survey on various noises and techniques for denoising the color image," International Journal of Application or Innovation in Engineering & Management (IJAIEM), 2 (11), 217–221.
40. Chang, Y., Yan, L., Fang, H., Zhong, S., Liao, W., 2018. Hsi-denet: Hyperspectral image restoration via convolutional neural network. IEEE Transactions on Geoscience and Remote Sensing 57 (2), 667–682.
41. Gholizadeh-Ansari, M., Alirezaie, J., Babyn, P., (2018) "Low-dose ct denoising with dilated residual network," In: 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, pp. 5117–5120.
42. Heinrich, M. P., Stille, M., Buzug, T. M., 2018. Residual u-net convolutional neural network architecture for low-dose ct denoising. Current Directions in Biomedical Engineering 4 (1), 297–300.
43. Jin, K. H., McCann, M. T., Froustey, E., Unser, M., (2017) "Deep convolutional neural network for inverse problems in imaging," IEEE Transactions on Image Processing 26 (9), 4509–4522.
44. Li, H., Yang, W., Yong, X., (2018) "Deep learning for ground-roll noise attenuation," In: SEG Technical Program Expanded Abstracts 2018. Society of Exploration Geophysicists, pp. 1981–1985.
45. Liu, W., Lee, J., (2019) "A 3-d atrous convolution neural network for hyperspectral image denoising," IEEE Transactions on Geoscience and Remote Sensing.
46. Lu, Y., Lai, Z., Li, X., Wong, W. K., Yuan, C., Zhang, D., (2018) "Low-rank 2-d neighborhood preserving projection for enhanced robust image representation," IEEE transactions on cybernetics 49 (5), 1859–1872.
47. Park, J. H., Kim, J. H., Cho, S. I., (2018) "The analysis of cnn structure for image denoising," In: 2018 International SoC Design Conference (ISOCC). IEEE, pp. 220–221.
48. Ren, W., Liu, S., Ma, L., Xu, Q., Xu, X., Cao, X., Du, J., Yang, M.-H., (2019) "Low-light image enhancement via a deep hybrid network," IEEE Transactions on Image Processing 28 (9), 4364–4375.
49. Su, Y., Lian, Q., Zhang, X., Shi, B., Fan, X., (2019) "Multi-scale cross-path concatenation residual network for poisson denoising," IET Image Processing.
50. Wang, T., Sun, M., Hu, K., (2017) "Dilated deep residual network for image denoising," In: 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI). IEEE, pp. 1272–1279.
51. Yu, A., Liu, X., Wei, X., Fu, T., Liu, D., (2018) "Generative adversarial networks with dense connection for optical coherence tomography images denoising," In: 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI). IEEE, pp. 1–5.
52. Hsu, C.-C., Lin, C.-W., (2017) "Cnn-based joint clustering and representation learning with feature drift compensation for large-scale image data," IEEE Transactions on Multimedia 20 (2), 421–429.
53. Li, Z., Zhang, Z., Qin, J., Zhang, Z., Shao, L., (2019) "Discriminative fisher embedding dictionary learning algorithm for object recognition," IEEE transactions on neural networks and learning systems.
54. Li, S., He, F., Du, B., Zhang, L., Xu, Y., Tao, D., 2019. Fast spatio-temporal residual network for video super-resolution. arXiv preprint arXiv:1904.02870.
55. Liu, P., Fang, R., 2017. Wide inference network for image denoising via learning pixel-distribution prior. arXiv preprint arXiv:1707.05414.
56. Meinhardt, T., Moller, M., Hazirbas, C., Cremers, D., (2017) "Learning proximal operators: Using denoising networks for regularizing inverse imaging problems," In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1781–1790.
57. Hongqiang, M., Shiping, M., Yuelei, X., Mingming, Z., (2018) "An adaptive image denoising method based on deep rectified denoising auto-encoder," In: Journal of Physics: Conference Series. Vol. 1060. IOP Publishing, p. 012048.
58. Cho, S. I., Kang, S.-J., (2018) "Gradient prior-aided cnn denoiser with separable convolution-based optimization of feature dimension," IEEE Transactions on Multimedia 21 (2), 484–493.
59. Yeh, R. A., Lim, T. Y., Chen, C., Schwing, A. G., Hasegawa-Johnson, M., Do, M., (2018) "Image restoration with deep generative models," In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp. 6772–6776.
60. Liu, P., Li, Y., El Basha, M. D., Fang, R., (2018) "Neural network evolution using expedited genetic algorithm for medical image denoising," In: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, pp. 12–20.
61. Tassano, M., Delon, J., Veit, T., (2019) "An analysis and implementation of the ffdnet image denoising method," Image Processing On Line 9, 1–25.
62. Jiao, J., Tu, W.-C., He, S., Lau, R. W., (2017) "Formresnet: Formatted residual learning for image restoration," In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 38–46.
63. Tao, L., Zhu, C., Xiang, G., Li, Y., Jia, H., Xie, X., (2017) "Llcnn: A convolutional neural network for low-light image enhancement," In: 2017 IEEE Visual Communications and Image Processing (VCIP). IEEE, pp. 1–4.

64. Han, Y., Ye, J. C., (2018) “Framing u-net via deep convolutional framelets: Application to sparse-view ct,” IEEE transactions on medical imaging 37 (6), 1418–1429.
65. Tian, C., Xu, Y., Zuo, W., (2020) “Image denoising using deep cnn with batch renormalization,” Neural Networks 121, 461–473.
66. Ioffe, S., (2017) “Batch renormalization: Towards reducing minibatch dependence in batch-normalized models,” In: Advances in neural information processing systems. pp. 1945–1953.
67. Chen, J., Hou, J., Chau, L.-P., (2018) “Light field denoising via anisotropic parallax analysis in a cnn framework.” IEEE Signal Processing Letters 25 (9), 1403–1407.
68. Jian, W., Zhao, H., Bai, Z., Fan, X., (2018) “Low-light remote sensing images enhancement algorithm based on fully convolutional neural network,” In: China High Resolution Earth Observation Conference. Springer, pp. 56–65.
69. Godard, C., Matzen, K., Uyttendaele, M., (2018) “Deep burst denoising,” In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 538–554.
70. Isogawa, K., Ida, T., Shiodera, T., Takeguchi, T., (2017) “Deep shrinkage convolutional neural network for adaptive noise reduction,” IEEE Signal Processing Letters 25 (2), 224–228.
71. Jaroensri, R., Biscarrat, C., Aittala, M., Durand, F., (2019) Generating training data for denoising real rgb images via camera pipeline simulation arXiv preprint arXiv:1904.08825.
72. LOO TIANG KUAN, L., (2017) Survey of deep neural networks in blind denoising using different architectures and different labels. Ph.D. thesis.
73. Soltanayev, S., Chun, S. Y., (2018) “Training deep learning based denoisers without ground truth data,” In: Advances in Neural Information Processing Systems. pp. 3257–3267.
74. Yang, J., Liu, X., Song, X., Li, K., (2017) “Estimation of signal-dependent noise level function using multicolumn convolutional neural network,” In: 2017 IEEE International Conference on Image Processing (ICIP). IEEE, pp. 2418–2422.
75. Zhang, F., Liu, D., Wang, X., Chen, W., Wang, W., (2018) “Random noise attenuation method for seismic data based on deep residual networks,” In: International Geophysical Conference, Beijing, China, 24-27 April 2018. Society of Exploration Geophysicists and Chinese Petroleum Society, pp. 1774–1777. https://doi.org/10.1190/IGC2018-435.
76. Si, X., Yuan, Y., (2018) “Random noise attenuation based on residual learning of deep convolutional neural network,” In: SEG Technical Program Expanded Abstracts 2018. Society of Exploration Geophysicists, pp. 1986–1990. https://doi.org/10.1190/segam2018-2985176.1.
77. Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo1, and Lei Zhang, “Toward Convolutional Blind Denoising of Real Photographs,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2019, Cornell University, arXiv:1807.04686v2 [cs.CV] 19 Apr 2019.

Received 07.04.22
Accepted 29.05.22